

REMARKS

Claims 1, 18, 25, and 41 have been amended to clarify the invention. Claims 1-2, 4-8 and 11-20, 22, 24-26, 28-32, and 34-41 remain pending.

The Examiner rejected claims 1-2, 4-8, 11-20, 22, 24-26, 28-32, and 34-41 under 35 U.S.C. §103(a) as being unpatentable over Li (U.S. Patent 5,643,125) and further in view of Dreszer (U.S. Patent 6,442,661). The Examiner's rejections are respectfully traversed as follows.

Claim 1 is directed towards a "method for assigning traffic buckets to a cache system." Claim 1 also requires the following steps:

a) when a new cache system starts up in a cache cluster having a plurality of cache systems among which a plurality of total buckets are to be allocated, distributing the total buckets among the cache systems, including the new cache system, so that a full bucket allocation is determined for the new cache system and assigning a portion of the full bucket allocation to the new cache system that was determined at start up;

b) periodically determining a load of the new cache system;

c) each time it is periodically determined that the new cache system is underloaded and buckets have not been previously shed from the new cache system, slowly assigning a portion of the full bucket allocation that was determined at startup without redistributing the total buckets among the cache systems, including the new cache system, unless the cache cluster is operating at a maximum load;

d) each time it is periodically determined that the new cache system is underloaded and buckets have been shed previously from the new cache system, slowly re-assigning a portion of the buckets that have been previously shed from the new cache system to the new cache system without redistributing the total buckets among the cache systems, including the new cache system, unless the cache cluster is operating at a maximum load; and

e) each time it is periodically determined that the new cache system is overloaded, shedding a portion of the buckets previously assigned to the new cache system without redistributing the total buckets among the cache systems, including the new cache system;

wherein each bucket portion corresponds to a portion of the total traffic being handled by the cache cluster.

Claims 18, 25, and 41 require mechanisms for performing the above recited steps. When a new cache system starts up, traffic buckets are distributed among the cache systems so that a full bucket allocation is determined for this new cache. In a slow start procedure, a portion of this determined full bucket allocation is slowly assigned to (or shed from) the new cache system (unless the cache cluster is at maximum load as claimed in claim 2). This slow start procedure also includes periodically monitoring the load of the new cache. When the new cache is overloaded or underloaded, only a portion of this full bucket allocation is assigned or shed without re-determining a new full bucket allocation, in the manner claimed. That is, during this slow start procedure, buckets are not redistributed to the cache systems so as to determine a new full bucket allocation.

This slow start procedure allows efficient distribution of allocated traffic buckets to a new cache system without quickly overwhelming the new cache system by assigning too much load at one time. At startup, portions of the full bucket allocation that was determined for the new cache system at startup are slowly assigned to (or shed from) the new cache system in a slow startup process. At startup, only a portion of the full bucket allocation is initially assigned to the new cache unless the cache cluster is at maximum load. Based on periodic monitoring of the new cache system's load, traffic bucket portions of the full allocation are then slowly assigned to this new cache system when buckets have not been previously shed. Portions of the buckets that have been previously shed are assigned when buckets have been shed previously. Since a full bucket allocation is not re-determined for the new cache in a re-distribution process, these mechanisms allow the new cache system to not be initially overwhelmed by its full bucket allocation. In embodiments of the present invention, the full bucket allocation is slowly assigned or slowly backed-off (shed) from the new cache system until the new cache system is capable of handling the full bucket allocation that was determined during distribution of the total buckets of the cache cluster at startup of the new cache system.

In contrast, the primary reference Li teaches distributing tables or data among a plurality of nodes at startup or during a load balancing process so as to determine data allocation for each node, where all of the data allocated to a particular node is immediately distributed to the particular node at one time without implementation of a slow start process in the manner claimed. For example, when a node becomes imbalanced, data is re-distributed to all the nodes to determine a new full data allocation for such imbalanced node and this new full data allocation is then assigned to the imbalanced node. See Col. 1, lines 50-52. Accordingly, Li fails to teach or suggest a slow start process, where portions of the full bucket allocation are slowly assigned or shed for a new cache system without performing re-distribution of such buckets, in the manner claimed.

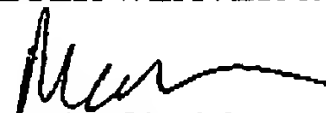
The secondary reference Dreszer also suffers from at least some of the above described deficiencies. In general, Dreszer is directed towards allocating memory blocks for use by various processes. See col. 2, line 66 through col. 3, line 1. Memory blocks or queues are allocated, resized and re-allocated to the various processes in various allocation and de-allocation procedures. See for example, col. 8, lines 27-55. Dreszer's allocation and de-allocation techniques directly result in making the new allocated or re-allocated block or queue available to a process for use by such process. See Col. 3, Lines 49-60: "memory manager 30, provides rapid memory allocation and de-allocation, reduced memory fragmentation, maximizes the amount of memory available for a cache...while optimizing the amount of memory available for other uses, and manages the competition for different memory uses..." (emphasis added). Col. 8, Lines 27-55 appears to describe methods for altering the allocated memory block size and then returning this new allocated memory block to the user. See Lines 53-55: "retuning the memory address immediately following the existing allocation header 98 to the operating system requester." In other words, the full allocated memory (or full re-allocated memory) is assigned to the process for its use, as opposed to slowly assigning portions of this full allocated (or full re-allocated) memory to such process. In sum, Dreszer fails to teach or suggest a slow start process, where portions of the full bucket allocation are slowly assigned to a new cache system without performing re-distribution of such buckets, in the manner claimed.

For the forgoing reasons, it is respectfully submitted that claims 1, 18, 25, and 41 are patentable over the cited references.

The Examiner's rejections of the dependent claims are also respectfully traversed. However, to expedite prosecution, all of these claims will not be argued separately. Claims 2, 4-8, 11-17, 19-20, 22, 24, 26, 28-32, and 34-40 each depend directly from independent claims 1, 18, or 25 and, therefore, are respectfully submitted to be patentable over cited art for at least the reasons set forth above with respect to claims 1, 18, and 25. Further, the dependent claims require additional elements that when considered in context of the claimed inventions further patentably distinguish the invention from the cited art.

Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP



Mary R. Olynick
Reg. 42,963

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100

Application No.: 09/608,549
CISCP151/3223